

---

## PROYECTO 2

---

202004765 – Javier Alejandro Gutierrez de León

### Resumen

Se explican los conceptos necesarios para la comprensión de las tecnologías utilizadas para la realización de este proyecto, además de la forma en que se implementaron para lograr establecer la lógica de ejecución del programa para lograr obtener las simulaciones además de los 3 tipos de reportes que se utilizaron para informar al usuario acerca del proceso en simulación.

Por ultimo se presentan los diagramas de clases y el método del árbol que se implementaron en el programa como métodos de abstracción para comenzar a plantear el problema para darle una solución más fácilmente familiarizándose con el problema sin realizar código sin sentido

### Palabras clave

**Cola:** es una estructura de datos, caracterizada por ser una secuencia de elementos

**Hilo:** es simplemente una tarea que puede ser ejecutada al mismo tiempo que otra tarea.

**Simulación:** es el artificio contextual que hace referencia a la investigación de una hipótesis o un conjunto de hipótesis de trabajo

### Abstract

*The concepts necessary to understand the technologies used to carry out this project are explained, as well as the way in which they were implemented to establish the program execution logic to achieve the simulations in addition to the 3 types of reports that were used. to inform the user about the process under simulation.*

*Finally, the class diagrams and the tree method that were implemented in the program are presented as abstraction methods to begin to pose the problem to give it a solution more easily becoming familiar with the problem without making senseless code*

### Keywords

**Queue:** it is a data structure, characterized by being a sequence of elements

**Thread:** it is simply a task that can be executed at the same time as another task.

**Simulation:** it is the contextual artifice that refers to the investigation of a hypothesis or a set of working hypotheses

## Introducción

Utilizando las estructuras de datos abstractas, se realizó un programa para simular el proceso de ensamblaje de productos descritos en un archivo XML según las configuraciones en la máquina que se utilizará para este fin, siendo ejecutado en tiempo real los diferentes procesos mostrándose en una lista el proceso que se está llevando.

Para de ultimo poder crear reportes en formato XML y HTML acerca del proceso que se llevó a cabo para la simulación seleccionada, también creando imágenes de la cola de componentes utilizada para cada producto que se simuló.

## Desarrollo del tema

**TDA:** En ciencias de la computación un tipo de dato abstracto o tipo abstracto de datos es un modelo matemático compuesto por una colección de operaciones definidas sobre un conjunto de datos para el modelo.

**Cola:** Una cola es una estructura de datos, caracterizada por ser una secuencia de elementos en la que la operación de inserción push se realiza por un extremo y la operación de extracción pull por el otro. También se le llama estructura FIFO, debido a que el primer elemento en entrar será también el primero en salir.

**Lista enlazada simple:** es una estructura de datos en la que cada elemento apunta al siguiente. De este modo, teniendo la referencia del principio de la lista podemos acceder a todos los elementos de la misma.

**Expresiones Regulares:** son patrones que se utilizan para hacer coincidir combinaciones de caracteres en cadenas.

**Autómata Finito:** es un modelo computacional que realiza cálculos en forma automática sobre una entrada para producir una salida. Este modelo está conformado por un alfabeto, un conjunto de estados finito, una función de transición, un estado inicial y un conjunto de estados finales.

**XML:** es un lenguaje de marcado similar a HTML. Significa Extensible Markup Language (Lenguaje de Marcado Extensible) y es una especificación de W3C como lenguaje de marcado de propósito general. ... El propósito principal del lenguaje es compartir datos a través de diferentes sistemas, como Internet.

**HTML:** (Lenguaje de Marcas de Hipertexto, del inglés HyperText Markup Language) es el componente más básico de la Web. Define el significado y la estructura del contenido web.

**CSS:** (Hojas de estilo en cascada), es un lenguaje de diseño gráfico para definir y crear la presentación de un documento estructurado escrito en un lenguaje de marcado.

**Hilo:** es simplemente una tarea que puede ser ejecutada al mismo tiempo que otra tarea.

Para la realización del proyecto se utilizaron los conceptos planteados anteriormente siendo empleados para los principales módulos y procedimientos para establecer los tiempos de realización de los productos y la configuración de las máquinas. Siendo el XML utilizado como archivos de entrada para darle configuración a la máquina de ensamblaje, así como la simulación de los elementos que se quieren realizar para una prueba, utilizando la librería de “xml etree” de Python para la lectura de estos archivos.

Continuando con pasar la lista de componentes obtenida por un Autómata finito para analizar la expresión regular establecida para obtener línea y componente necesario, almacenándolos en una cola para utilizarlos mas adelante. Estableciendo ambos archivos en una lista de objetos con los datos de simulación, nombre de objeto y componentes necesarios. Comenzando a ejecutar la simulación se comenzará por el primer objeto en la lista de obtenida del archivo de simulación utilizando la cola de componentes para comenzar a ensamblar.

Durante la ejecución del programa que se realiza desde un hilo se podrá apreciar la actualización en tiempo real de los movimientos de los brazos al ensamblar. Al final se pueden generar graficas utilizando Graphviz para diagramar la cola de cada producto en la simulación, además de generar reportes en HTML y XML de los movimientos que se ejecutaron para ensamblar el producto.

### **Conclusiones**

Utilizar el lenguaje Python fue de gran ayuda, ya que al ser un lenguaje más simple se lograron realizar funciones complejas en pocas líneas de código, las cuales hubieran sido muchas más si se hubiese utilizado algún otro lenguaje de programación.

La utilización de la memoria dinámica con los TDA al principio parecía algo complicada, pero al entender bien estos se lograron implementar funciones que para aspectos prácticos de búsqueda y toma de decisiones con el algoritmo implementado fue más práctico que utilizar las listas que ya trae Python ya que con estas se tenían que buscar los datos de arriba y abajo tomando en cuenta la cantidad de datos que debería haber por fila.

La manipulación de archivos XML por medio de cElementTree fue muy sencillo ya que con las

funciones que este posee fue fácil extraer la información dentro del archivo, ahorrando la interpretación de línea por línea y la eliminación de caracteres que no fueran relevantes que se debería realizar si no utilizáramos esta librería.

La utilización de Graphviz fue complicado ya que esta herramienta tiene varios usos, por lo que la confusión por parte paginas no oficiales dificultó su uso, pero al leer la documentación se logró implementar la estructura para generar la gráfica de las colas para cada producto.

La utilización de HTML para generar tablas de la simulación dio un buen resultado siendo utilizado Bootstrap para mejorar la apariencia grafica de estas.

### **Referencias bibliográficas**

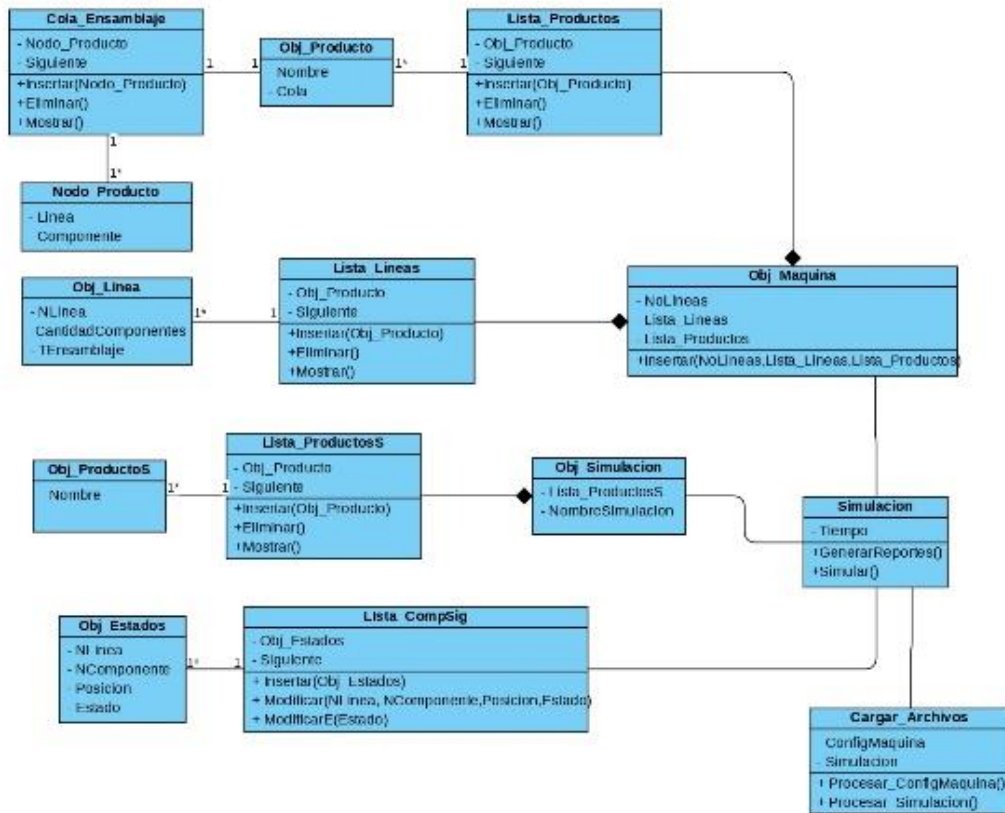
Academy, P. L. (2020). Python para Principiantes: 2 Libros en 1: Programación de Python para principiantes + Libro de trabajo de Python. Independently Published.

Budgen, D. (2007). Software Design (English Edition) (2.a ed.). Addison Wesley.

Creately. (2021, 15 enero). La Guía Fácil de los Diagramas de Despliegue UML. Blog de Creately. <https://creately.com/blog/es/diagramas/tutorial-de-diagrama-de-despliegue/>

## Anexos

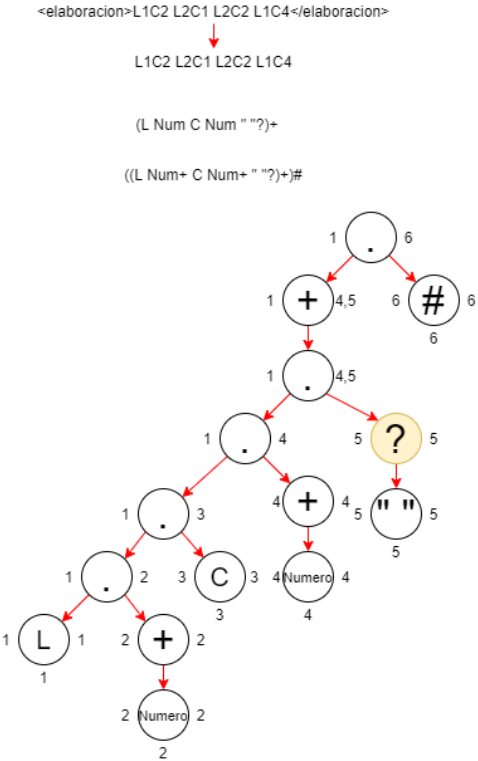
Visual Paradigm Online Free Edition



Visual Paradigm Online Free Edition

Figura 1. Diagrama de Clases implementado en el proyecto.

Fuente: elaboración propia, 2021.



Hoja	Terminal	Siguiente
1	L	2
2	Numeros	2,3
3	C	4
4	Numeros	4,5,6
5	" "	1
6	#	-

Estados	L	Numeros	C	" "	#
S0	S1	-	-	-	-
S1	-	S1	S2	S0	S3
S2	-	S1	-	-	-
S3	-	-	-	-	-

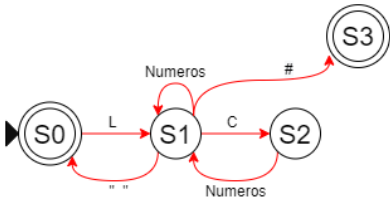


Figura 2. Método del árbol para obtener autómata implementado para reconocer lista y componente

Fuente: elaboración propia, 2021.